CS172 Computer Vision I:
Depth Map Prediction from a Single Image using a Multi-Scale Deep Network

Li Wuchen
2018533114
liwch@shanghaitech.edu.cn

Abstract

Depth prediction is an essential task in understanding the geometric relations for a scene. The traditional technique using multi-view images is well developed in predicting depth of a scene. In this report, we consider the problem of how to draw the depth information from a single RGB image. Finding depth relations from a single image is not straightforward, so we employ a multi-column deep network to address such task as proposed by Eigen et. al. [1]. The deep net includes two stacks: one makes a coarse prediction incorporating global information of the entire image, and the other refines the prediction based on local features. The model is trained and experimented with NYU Depth V2 dataset.

1. Introduction

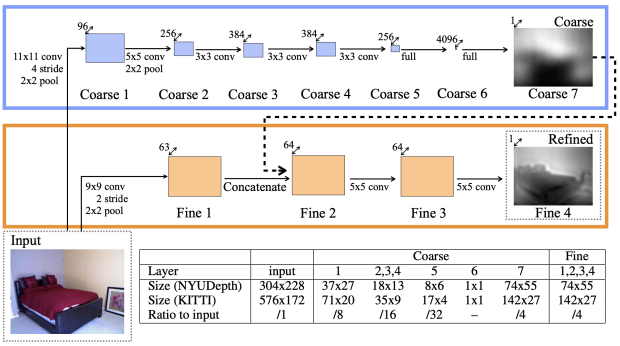
Depth prediction has been widely explored in computer vision. Traditional methods take multiple sights of same scene to predict the depth, e.g. given pictures shot by two calibrated cameras. With accurate image point correspondence and parameters of camera, the depth of scene can be recovered deterministically. But more often, monocular cases arises in practice, like pictures post on social media. Predicting depth from a single image takes more cues, e.g. object size, relation between objects, perspective. Such problem is tricky as given any scene in the world, infinite pictures can be created and given a picture, infinite scenes can be created. But only a few of scenes are acceptable when it comes to the real world. Thus, we can address this problem by introducing a scale-invariant error instead of more common scale-dependent errors. It allows us to focus on the spatial relations inside the image rather than the general relations. In this report, I reimplement the work of Eigen et. al. I directly regress on the depth using a neural net based on Keras. The net contains two components: one that first estimates the global structure of the scene, then a second that refines it using local information.

2. Model Implementation

In this report, I choose Keras, based on tensorflow, as network back bone. It contains friendly api and is easily to read and code.

2.1. Model Architecture

The net contains two component stacks as shown in Figure 1. First, a coarse net predicts the depth of scene at a global level. This is then refined with local features by a fine-scale net. The local network will edit the global prediction incorporate with fine-scale details.



Layer	input	Coarse					Fine
		1	2,3,4	5	6	7	1,2,3,4
Size (NYUDepth)	304x228	37x27	18x13	8x6	1x1	74x55	74x55
Size (KITTI)	576x172	71x20	35x9	17x4	1x1	142x27	142x27
Ratio to input	/1	/8	/16	/32	–	/4	/4

Figure 1. Model structure

2.2. Global Coarse-Scale Network

Model structure is implemented as same as the figure shown above. Coarse-scale network contains five feature extraction layers of convolution, a pooling, followed by two fully connected layers. The input is a 320x240 RBG image and the output is a 80x60 depth map, which means the final output is at 1/4-resolution compared to the input size.

2.3. Local Fine-Scale Network

This network align the coarse predictions with the local details. The last fully connected layer predicts the depth. The coarse network is trained first and it is fixed when the fine network is under training.

2.4. Model parameters

- Epochs = 30
- learning rate = 0.1
- momentum = 0.9
- input size = 320x240x3
- output size = 80x60
- Batch Size = 32

3. Dataset

I use NYU Depth V2 dataset for training and testing. This dataset contains 1449 indoor RGB images with a resolution of 640x480 and all images are labelled with depth map. I use *h5py* package to read dataset provided by NYU and store them in form of nparray. In order to fit the model mentioned in Eigen's work, images are downsampled for once, and depth maps are downsampled for three times. Then data are split to two set: training set containing 1200 images and testing set containing 249 images. As the data set takes too much space in memory, I store and only focus on the downsampled data.

4. Results

Eigen use a much larger data set for training. In this report the model is just trained with a smaller scale dataset for 1 hour. So the result is impossible to be as good as Eigen's result but can predict the depth well to a certain extent.

4.1. NYU Depth

Both coarse-scale network and fine-scale network are trained for 30 echoes.Two networks are fit with the downsampled final depth map. Fine-scale network apparently shows a better performance in prediction accuracy compared with coarse-scale network. The result of training is shown below.

	loss	accuracy	val loss	val accuracy
coarse	0.0692	0.0192	0.0507	0.0251
fine	0.8591	0.0555	0.8602	0.0511

With model evaluation provided by Keras model, the result shows even if the fine-scale network has a larger loss than coarse-scale network but also achieve a much better prediction result.

4.2. Result Display

Test with indoor images provided by NYU Depth V2 dataset.

Test with any indoor image.

Depth prediction map:

	loss	accuracy
coarse	0.0853	0.0237
fine	1.0396	0.0964

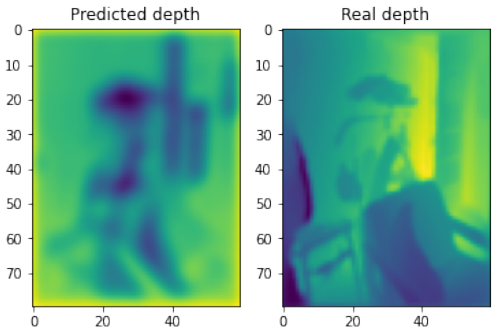


Figure 2. Test imge 1

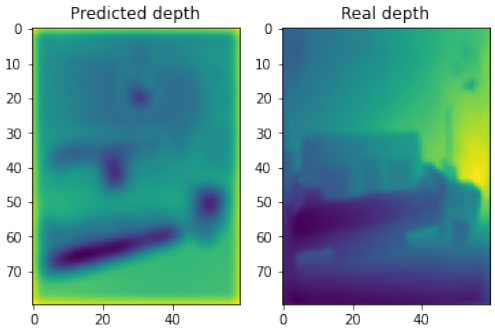


Figure 3. Test imge 2

Figure 4. RGB image after rescaling

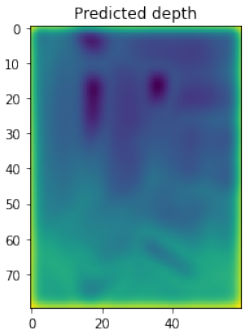


Figure 5. Depth prediction map

5. Discussion

Eigen et. al. provided a good way to estimate depth from a single image. By combining information from global view and local view, the model can perform quite well. It is a good project as a first try of getting familiar with CNN. But due to time limit, there are some perspective can be considered as further improvement. First, as I do not have enough time to train my model, the epochs is set to 30 which could have been 150. And for this network, increasing the number of layer may result in a better solution and as the depth of net increasing, dropout layers can be added to reduce overfitting.

References

[1] Eigen, David, Christian Puhrsch, and Rob Fergus. "Depth map prediction from a single image using a multi-scale deep network." Advances in neural information processing systems. 2014.